



Ti metto una sintesi "non corta", ma ancora leggibile, delle opzioni + alcuni riferimenti che puoi citare. Sono tutti lavori che usano CNN o deep learning su ECG, e che puoi usare per giustificare le diverse scelte di rappresentazione.

1. CNN 1D su serie temporale multicanale (approccio standard)

Idea generale

- L'ECG è visto come **serie temporale multivariata**:
 - tensore $X \in \mathbb{R}^{T \times C}$
 - T = numero di campioni nel tempo (es. 2500–5000)
 - C = numero di canali (lead, tipicamente 1, 2 o 12).
- Si applicano **Conv1D lungo il tempo**, in cui i canali sono feature parallele:
 - kernel (k_t) che scorrono su T ;
 - combinano subito informazioni temporali e tra derivazioni.
- Architettura tipica:
 - stack di Conv1D → BatchNorm → ReLU → MaxPool1D;
 - eventualmente GlobalAveragePooling1D o Flatten → Dense per classificazione.

Questo è il paradigma più diffuso per ECG, e funziona molto bene sia su singola derivazione sia su 12 derivazioni.

Come vengono trattati i canali

- Per 12-lead ECG:
 - l'input è tipicamente ($T, 12$);
 - la prima convoluzione combina le 12 derivazioni come 12 feature.
- Non c'è quasi mai un pre-processing "manuale" dei canali (tipo pesi a mano per ogni lead); si lascia alla rete il compito di imparare automaticamente quali lead sono più informativi.

Esempi di architetture

1. Rajpurkar et al., 2017 (arXiv) / 2019 (Nature Medicine)

- Arrhythmia detection su ECG a singola derivazione (ma la logica CNN 1D è la stessa per più canali).
- **Riferimenti:**
 - P. Rajpurkar et al., "Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks", arXiv:1707.01836, 2017.
 - A. Y. Hannun et al., "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network", *Nature Medicine*, 2019.

2. Strodthoff & Strodthoff, 2021 – Deep Learning for 12-lead ECG

- Usano CNN 1D su 12-lead ECG per diagnosi cardiache.
- Discussione esplicita sulla rappresentazione multicanale e su architetture 1D vs 2D.
- **Riferimento:**
 - N. Strodthoff, C. Strodthoff, "Deep learning for ECG analysis: Benchmarks and insights from PTB-XL", *IEEE Journal of Biomedical and Health Informatics*, 25(5), 2021.

3. Ribeiro et al., 2020 – Mortalità e diagnosi da 12-lead

- CNN 1D su 12-lead ECG per predire esiti clinici (mortalità, condizioni cardiache).
- Input: finestra fissa multicanale; la rete è una CNN 1D relativamente "classica".
- **Riferimento:**
 - A. L. P. Ribeiro et al., "Automatic diagnosis of the 12-lead ECG using a deep neural network", *Nature Communications*, 11, 2020.

Come puoi collegarlo al tuo progetto

- Il tuo input 5000×12 è un esempio perfetto di serie temporale multicanale.
- Una CNN 1D che lavora su $(T, 12)$ è coerente con la maggior parte dei lavori recenti su 12-lead ECG.
- Il fatto che i 12 canali siano trattati come componenti parallele è standard; puoi citare Strodthoff & Strodthoff (PTB-XL) e Ribeiro et al. per 12-lead.

2. Architetture multi-branch / multi-lead (es. 6+6)

Idea generale

- L'idea è **non trattare tutti i canali allo stesso modo**, ma sfruttare la struttura dei lead:
 - limb leads (I, II, III, aVR, aVL, aVF),
 - precordiali (V1–V6).
- Si costruiscono **più branch** (rami) nella rete:
 - **Branch 1**: input $X^{(1)} \in \mathbb{R}^{T \times 6}$ (derivazioni degli arti),
 - **Branch 2**: input $X^{(2)} \in \mathbb{R}^{T \times 6}$ (precordiali).
- Ogni branch ha un encoder CNN 1D (stessa architettura, pesi diversi):
 - produce embedding z_1 e z_2 .
- Poi si fa una **fusione**: concatenazione $[z_1, z_2]$, oppure somma, oppure attention.

Questa filosofia si adatta molto bene al tuo caso, dove i 12 canali arrivano da due blocchi da 6 lead con un leggero sfasamento temporale.

Motivazioni in letteratura

- I pattern elettrofisiologici sulle precordiali (V1–V6) sono diversi da quelli sugli arti.
- Multi-branch permette:
 - di catturare pattern specifici per ogni gruppo di derivazioni;
 - di combinare successivamente queste informazioni in modo più flessibile.
- È anche una forma di **multimodal learning "intra-ECG"** (due viste diverse dello stesso cuore).

Esempi / riferimenti

In letteratura i multi-branch sono spesso presentati in due versioni:

1. Multi-lead specific feature extraction

- Alcuni lavori definiscono branch separati per sottoinsiemi di lead o per lead derivati (es. DII, V5, V2) perché particolarmente informativi per alcune diagnosi.
- Esempio:
 - Y. Wang et al., "Deep learning for ECG classification: where are we now?", *Computers in Biology and Medicine*, 2021 (review che cita diverse architetture multi-branch e multi-lead).

2. Multi-branch per feature diverse (raw + trasformazioni)

- Alcune architetture hanno un branch per il segnale raw e un branch per feature ingegnerizzate (onde P-QRS-T, intervalli, ecc.), poi fusi in uno strato comune.
- Esempio:
 - J. Li et al., "Multi-scale convolutional neural network for ECG classification", *IEEE Access*, 2018 – usa branch con kernel diversi (multi-scale), concetto simile ai multi-branch per diversi canali.

Non sempre la divisione è esattamente 6+6 lead come nel tuo dataset, ma il principio è lo stesso: **rami separati che processano viste diverse dell'ECG**, poi fusi in un bottleneck.

Collegamento al tuo progetto

- Il tuo dataset ha due blocchi fisicamente separati (6+6 lead, 5 s ciascuno, con lag).
- Una multi-branch 1D CNN:
 - branch 1: blocco arti,
 - branch 2: blocco precordiali,
 - bottleneck comune → classificatore, è perfettamente in linea con la traccia del progetto e con l'idea multi-branch della letteratura.
- Puoi motivare dicendo che:
 - "seguiamo il paradigma multi-branch descritto in [Wang 2021, Li 2018], abbiamo progettato due encoder separati per i blocchi limb/precordiali, che catturano pattern morfologici specifici prima di essere fusi in uno spazio di rappresentazione comune".

3. ECG come "immagine" 2D (Conv2D)

Idea generale

- Invece di trattare l'ECG come serie 1D con C canali, lo si rappresenta come una **immagine 2D**:
 - dimensione verticale: lead (12 righe),
 - dimensione orizzontale: tempo (T colonne),
 - valore in ciascun pixel: ampiezza del segnale.

- Si ottiene quindi una matrice $12 \times T$ che può essere vista come immagine "grayscale":
 - input shape: $(1, 12, T)$ oppure $(12, T, 1)$ a seconda del framework.

- Si usano CNN 2D standard:
 - kernel 2D, ad es. (3×3) o (3×7) ,
 - pooling lungo il tempo (e talvolta lungo l'asse delle derivazioni).

Motivazioni

- Le conv 2D possono:
 - catturare pattern **tempo-lead** con filtri 2D (pattern simultanei su più derivazioni),
 - sfruttare l'ordinamento delle derivazioni per modellare vicinanze "spaziali".
- Consente di riusare architetture da computer vision (ResNet, VGG, EfficientNet, ecc.), eventualmente pre-addestrate e riadattate.

Esempi / riferimenti

1. Xia et al., 2018 – 2D CNN su "ECG images"

- Convertono segmenti di ECG in immagini 2D e addestrano CNN 2D per classificare aritmie.
- **Riferimenti:**
 - Y. Xia et al., "Detecting atrial fibrillation by deep convolutional neural networks", *Computers in Biology and Medicine*, 93, 2018.

2. Acharya et al., 2017 – CNN 2D su trasformazioni dell'ECG

- Usano trasformazioni (es. wavelet) per generare scalogrammi e poi CNN 2D.

- **Riferimenti:**
 - U. R. Acharya et al., "A deep convolutional neural network model to classify heartbeats", *Computers in Biology and Medicine*, 89, 2017.

3. Strodthoff & Strodthoff, 2021 (di nuovo)

- Confrontano 1D vs 2D CNN su PTB-XL (12-lead);
- mostrano che entrambe le rappresentazioni sono valide, con trade-off in termini di complessità e prestazioni.

Collegamento al tuo progetto

- Potresti costruire una matrice $12 \times T$ (ad esempio $T=3000$) per ciascun ECG e usare Conv2D.
- È un approccio "più pesante" e meno immediato, ma citabile e già utilizzato su 12-lead.
- Se non hai molto tempo, è più sicuro restare sulla CNN 1D; ma nella relazione puoi comunque menzionare questa opzione come alternativa presente in letteratura.

4. Trasformazioni tempo-frequenza (spectrogrammi / scalogrammi) + CNN 2D

Idea generale

- Invece del segnale nel dominio del tempo, si usa una rappresentazione **tempo-frequenza**:
 - STFT (Short-Time Fourier Transform) → spettrogramma;
 - CWT (Continuous Wavelet Transform) → scalogramma.

- Per una singola derivazione si ottiene una matrice $F' \times T'$ (frequenza × tempo).

- Per 12 derivazioni, diverse scelte:
 - stack delle 12 immagini come 12 canali (simile a R/G/B ma 12 canali),
 - oppure concatenazione verticale delle 12 mappe.

Queste immagini vengono date in input a **CNN 2D**.

Motivazioni

- Le conv 2D possono:
 - catturare pattern **tempo-lead** con filtri 2D (pattern simultanei su più derivazioni),
 - sfruttare l'ordinamento delle derivazioni per modellare vicinanze "spaziali".
- Consente di riusare architetture da computer vision (ResNet, VGG, EfficientNet, ecc.), eventualmente pre-addestrate e riadattate.

Esempi / riferimenti

1. Ribeiro et al., 2020 (Nature Communications)

- Nella rete per diagnosi e predizione di mortalità con 12-lead ECG, considerano anche dati clinici (età, sesso ecc.) come feature aggiuntive (anche se in alcune versioni sono usati per valutare performance per sottogruppi).

- Questo approccio è standard nel **multimodal learning** (segnale + tabella).

2. Acharya et al., 2017 – CNN 2D su trasformazioni dell'ECG

- Usano trasformazioni (es. wavelet) per generare scalogrammi e poi CNN 2D.

- **Riferimenti:**
 - U. R. Acharya et al., "A deep convolutional neural network model to classify heartbeats", *Computers in Biology and Medicine*, 89, 2017.

3. Strodthoff & Strodthoff, 2021 (di nuovo)

- Confrontano 1D vs 2D CNN su PTB-XL (12-lead);
- mostrano che entrambe le rappresentazioni sono valide, con trade-off in termini di complessità e prestazioni.

Collegamento al tuo progetto

- È un'opzione più avanzata (richiede calcolo di STFT/CWT, scelta delle finestre, parametri, ecc.).
- Puoi citarla come alternativa dalla letteratura quando discuti possibili estensioni del lavoro:

- "In alternativa alla CNN 1D sul segnale raw, diversi autori propongono di convertire l'ECG in mappe tempo-frequenza (STFT o CWT) da processare con CNN 2D [Acharya 2017, Kiranyaz 2021], ma per semplicità e coerenza con la durata limitata dei tracciati abbiamo scelto...".

5. Integrazione ECG + dati tabellari (branch aggiuntivo MLP)

Idea generale

- Invece del segnale nel dominio del tempo, si usa una rappresentazione **tempo-frequenza**:
 - STFT (Short-Time Fourier Transform) → spettrogramma;
 - CWT (Continuous Wavelet Transform) → scalogramma.

- Per una singola derivazione si ottiene una matrice $F' \times T'$ (frequenza × tempo).

- Per 12 derivazioni, diverse scelte:
 - stack delle 12 immagini come 12 canali (simile a R/G/B ma 12 canali),
 - oppure concatenazione verticale delle 12 mappe.

Queste immagini vengono date in input a **CNN 2D**.

Motivazioni

- Alcune patologie generano pattern frequenziali caratteristici (es. pattern ritmici di aritmie).
- Le CNN 2D sono molto efficaci su immagini e mappe tempo-frequenza.
- Permette di sfruttare pretraining su immagini (resnet, ecc.), benché il dominio sia diverso.

Esempi / riferimenti

1. Kiranyaz et al., 2018 – 2D CNN su "ECG images"

- Convertono segmenti di ECG in immagini 2D e addestrano CNN 2D per classificare aritmie.

- **Riferimenti:**
 - Y. Xia et al., "Detecting atrial fibrillation by deep convolutional neural networks", *Computers in Biology and Medicine*, 93, 2018.

2. Acharya et al., 2017 – CNN 2D su trasformazioni dell'ECG

- Usano trasformazioni (es. wavelet) per generare scalogrammi e poi CNN 2D.

- **Riferimenti:**
 - U. R. Acharya et al., "A deep convolutional neural network model to classify heartbeats", *Computers in Biology and Medicine*, 89, 2017.

3. Strodthoff & Strodthoff, 2021 (di nuovo)

- Confrontano 1D vs 2D CNN su PTB-XL (12-lead);
- mostrano che entrambe le rappresentazioni sono valide, con trade-off in termini di complessità e prestazioni.

Collegamento al tuo progetto